Computer Science

ERROR CORRECTION IN A TRAINING CORPUS FOR A PART-OF-SPEECH TAGGER

Maria Enderton, Macalaster College, etc.
Andrew Stout, Swarthmore College, etc.
Scott Thede (*), DePauw University, etc.

One fundamental task in Natural Language Processing is part-of-speech tagging. Part-of-speech (POS) tagging is the task of determining how a word is being used in a given sentence, and "tagging" the word accordingly. It is important develop accurate methods for automatic POS tagging because POS tagging is a foundational process upon which many further processing tasks (such as parsing or translation) build and depend. There are various approaches to POS tagging, but nearly all current successful methods rely on training the tagger program on a large set of previously tagged sentences, called a corpus. These approaches are generally rather successful, but they are limited by the reliability of training corpus; a tagger can be no better than the corpus from which it learns. Since it is very difficult to compile a corpus of practical size (tens of thousands of sentences at the least) with more than about 96-97% tagging accuracy, this limitation imposes a ceiling on tagger performance. Our research is a first step toward breaking this ceiling by improving the accuracy of POS tagger training data.

The nature of this problem is such that the strengths of both humans and computers must be utilized. Clearly it is prohibitively impractical to hand-correct a corpus of millions of words. Moreover, any hand-correction of that magnitude is plagued by human inconsistency and fallibility, which is the source of no small portion of the present inaccuracies. Unfortunately, while task size and consistency are not problems for a computer, a corpus cannot be corrected by a computer if the computer does not know how to recognize an error. Therefor, to develop and evaluate automated methods of error correction, we had to first hand-correct a sample of 1000 sentences.

We developed a few different strategies for automatic error correction. One primary approach constructed a list of errors by identifying, in an uncorrected corpus sample, those ambiguous words (words tagged more than one way) for which one tag accounted for less than a threshold percentage of the total occurrences of the word. A second approach compared the hand-corrected and uncorrected versions of a corpus sample and generated a list of errors based on that comparison. In at least some variations on each approach a limited scope of context was also taken into account in constructing the list of errors.

We evaluated the performance of these programs using 'cross-validation' on the 1000-sentence sample of corpus we hand-corrected. Cross validation involves "training" the program in question on 90% of the sample and "testing" it on the other 10%, then repeating this until the program has been tested on the whole sample. For our purposes, "training" involved generating the list of errors, and "testing" was measuring how well the trained program then corrected a given sample to which the answers were known. We also measured the performance of some strategies trained on a larger (uncorrected) sample.

While the performance of the automated strategies indicated that significant further work is required before a computer can confidently be used to correct a corpus, we did achieve some encouraging results. The program that trains on both the corrected and uncorrected versions of a corpus sample performed significantly better than the "blind" approach. One result which suggested the utility of automated correction is that one trial of the second strategy identified three errors in the 1000-sentence sample which we humans did not find in our hand-correction. We believe that such an approach has the potential to be a useful tool to aid in human-performed hand-correction.

The largest limitation to the automated methods we developed was sparse data. The majority of errors in the corpus sample we corrected occurred in words which appeared too few times to be detected by a statistical approach to error-identification. Consequently, we recommend that further work on this topic address methods for detecting and correcting errors in sparse data.

Acknowledgement: NSF REU grant number 9911626 supported this work.